

BAB II

TINJAUAN PUSTAKA

2.1 Laboratorium Virtual

Virtual Lab adalah program komputer, di mana mahasiswa berinteraksi dengan percobaan menggunakan komputer melalui jaringan komputer (Stark, 2017). Melalui Virtual Lab, mahasiswa dapat melakukan simulasi praktikum, membaca materi, mengerjakan soal, melakukan diskusi dan mendapatkan evaluasi atas pekerjaannya.

Pertumbuhan internet dan teknologi komunikasi telah mempengaruhi sistem pendidikan dalam banyak hal. Media pengajaran misalnya, telah berubah dari komputer *offline* ke komputer berbasis web, dan dari media cetak ke digital. Cara komunikasi antara guru dan siswa telah berubah juga. Sebagai contoh, komunikasi dilakukan dengan hanya bertatap muka secara fisik, sedangkan sekarang bisa melalui surat elektronik, chat, dan kelas virtual.

Sehingga munculnya kebutuhan untuk memiliki kelas tanpa batasan waktu untuk belajar dan melakukan percobaan di mana saja dan kapan saja yang kemudian mendorong pembangunan laboratorium virtual. Menurut Alexio (2005), laboratorium virtual atau V-Lab diharapkan bisa menjadi cara yang murah untuk memberi kemudahan akses ke mahasiswa dan agar mereka memiliki pemahaman yang lebih baik dalam cara virtualisasi melalui antarmuka web / aplikasi atau *remote* perangkat sebenarnya (Stark, 2017).

Menurut Stark (2017), terdapat beberapa perbedaan antara virtual lab dan lab nyata, di mana masing-masing lab memiliki kelebihan dan kekurangannya, seperti yang dijelaskan pada tabel berikut :

Jenis Laboratorium	Kelebihan	Kekurangan
Nyata	<ul style="list-style-type: none"> • Menggunakan data real • Berinteraksi langsung dengan pembimbing • Dapat bekerja sama 	<ul style="list-style-type: none"> • Terbatas waktu dan tempat • Membutuhkan penjadwalan • Mahal • Memerlukan Pengawasan
Virtual	<ul style="list-style-type: none"> • Bagus untuk menjelaskan konsep • Tanpa batasan waktu dan tempat • Biaya murah 	<ul style="list-style-type: none"> • Tidak berinteraksi langsung dengan pembimbing • Tidak bisa bekerja sama

Tabel 2. 1 Perbandingan Lab Virtual dan Lab Nyata

(Sumber: Stark, 2017)

Keterampilan pemrograman memerlukan waktu latihan yang lama dan secara berkesinambungan. Penggunaan Virtual Lab dapat mendukung penulisan kode program tanpa perlu memasang banyak aplikasi untuk mengompilasi program sehingga dapat menghemat penyimpanan.

2.2 Sistem

Karakteristik sebuah sistem adalah terdiri dari bagian-bagian yang saling berkaitan dan beroperasi untuk mencapai suatu tujuan. Sebuah sistem bukanlah seperangkat unsur yang tersusun secara tidak teratur, namun sistem terdiri dari unsur yang dapat dikenal untuk saling melengkapi karena memiliki maksud, tujuan dan sasaran tertentu.

Jogiyanto (2005) berpendapat bahwa sistem adalah suatu kesatuan yang terdiri dari dua atau lebih komponen atau sub sistem yang berinteraksi untuk mencapai suatu tujuan.

2.2.1 Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang, di antaranya adalah sebagai berikut :

1. Sistem Abstrak (*abstract system*) dan Sistem Fisik (*physical system*)

Sistem abstrak adalah sistem yang berisi gagasan atau konsep. Sedangkan sistem fisik adalah sistem yang secara fisik dapat dilihat.

2. Sistem Deterministik (*deterministic system*) dan Sistem Probabilistik (*probabilistic system*)

Sistem deterministik adalah suatu sistem yang operasinya dapat diprediksi secara tepat. Sedangkan sistem probabilistik adalah sistem yang tidak dapat diramal dengan pasti karena mengandung unsur *probabilistic*.

3. Sistem Tertutup (*closed system*) dan Sistem Terbuka (*open system*)

Sistem tertutup adalah sistem yang tidak bertukar materi, informasi, atau energi dengan lingkungan. Sedangkan sistem terbuka adalah sistem yang berhubungan dengan lingkungan dan dipengaruhi oleh lingkungan.

4. Sistem Alamiah (*natural system*) dan Sistem Buatan manusia (*human made system*)

Sistem alamiah adalah sistem yang terjadi karena alam (tidak dibuat oleh manusia). Sedangkan sistem buatan manusia adalah sistem yang dibuat oleh manusia.

5. Sistem Sederhana dan Sistem kompleks

Berdasarkan tingkat kerumitannya, sistem dibedakan menjadi sistem yang sederhana dan sistem yang kompleks.

2.2.2 Karakteristik Sistem

Menurut Jogiyanto (2005), suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen, batas sistem, lingkungan luar sistem, penghubung dan sasaran, diantaranya:

1. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu sub sistem atau bagian-bagian dari sistem. Setiap sub sistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batas Sistem

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan.

3. Lingkungan Luar Sistem

Lingkungan luar dari suatu sistem adalah apa pun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Sedang lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

4. Sub sistem

Bagian dari Sistem yang beraktivitas dan berinteraksi satu sama lain untuk mencapai tujuan dengan sasarannya masing-masing.

5. Penghubung Sistem

Penghubung merupakan media penghubung antara satu sub sistem dengan sub sistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu sub sistem ke sub sistem yang lainnya. Keluaran dari satu sub sistem akan menjadi masukan untuk sub sistem yang lainnya dengan melalui penghubung.

6. Masukan Sistem

Masukan (*input*) adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). Masukan perawatan adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Masukan sinyal adalah energi yang diproses untuk didapatkan keluaran.

7. Keluaran Sistem

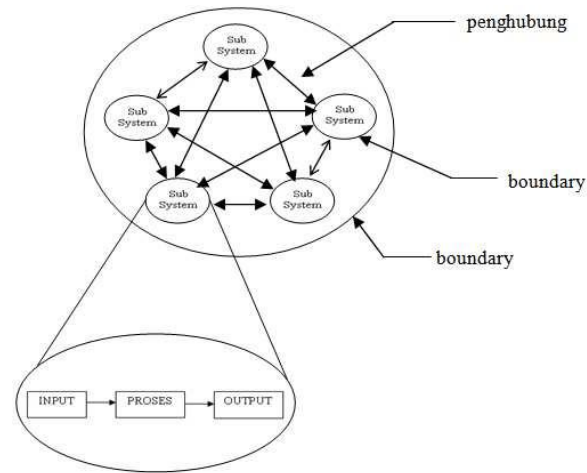
Keluaran (*output*) adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Misalkan untuk sistem komputer, panas yang dihasilkan adalah keluaran yang tidak berguna dan merupakan hasil sisa pembuangan, sedangkan informasi adalah keluaran yang dibutuhkan.

8. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan mengubah masukan menjadi keluaran.

9. Sasaran Sistem

Suatu sistem pasti mempunyai tujuan atau sasaran kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan berguna. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.



Gambar 2. 1 Karakteristik Sistem

(Sumber: Jogiyanto, 2005)

Bentuk umum dari sistem terdiri dari atas masukan (*input*), proses, dan keluaran (*output*). Dalam bentuk umum sistem ini biasa melakukan satu atau lebih masukan yang akan diproses dan menghasilkan keluaran sesuai dengan yang direncanakan sebelumnya.

2.2.3 Perancangan Sistem

Menurut Jogiyanto (2005), Desain Sistem adalah penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi.

Dari penjelasan tersebut maka penulis dapat mengambil kesimpulan bahwa perancangan (Desain) adalah proses penerjemahan kebutuhan pemakai informasi dengan melakukan penggambaran, perencanaan, dan pembuatan sketsa ke dalam kesatuan yang berfungsi.

2.2.3.1 Tahapan Perancangan

Tahapan utama dalam proses pengembangan sistem informasi adalah sebagai berikut:

1. Investigasi Sistem

Manfaat dari fase penyelidikan ini adalah untuk menentukan problem-problem atau kebutuhan yang timbul. Hal itu memerlukan pengembangan sistem secara menyeluruh ataukah ada usaha lain yang dapat dilakukan untuk memecahkannya. Salah satu alternatif jawabannya mungkin saja merupakan suatu keputusan untuk tidak melakukan perubahan apa pun terhadap sistem yang berjalan. Dengan kata lain sistem yang ada tetap berjalan tanpa perlu perubahan maupun pembangunan sistem yang baru.

2. Analisis Sistem

Tahap analisis bertitik tolak pada kegiatan-kegiatan dan tugas-tugas di mana sistem yang berjalan dipelajari lebih mendalam, konsepsi dan usulan dibuat untuk menjadi landasan bagi sistem yang baru yang akan dibangun. Pada akhir tahap ini separuh kegiatan dari usaha pengembangan sistem informasi telah diselesaikan. Salah satu tujuan terpenting pada tahap ini adalah untuk mendefinisikan sistem berjalan.

3. Desain Sistem

Pada tahap ini sebagian besar kegiatan yang berorientasi ke komputer dilaksanakan. Spesifikasi perangkat keras dan perangkat lunak (*Hardware/Software*) yang telah disusun pada tahap sebelumnya ditinjau kembali dan disempurnakan. Rencana pembuatan program dilaksanakan dan juga testing programnya. Latihan bagi para pemakai sistem dimulai. Pada akhirnya dengan berpartisipasi penuh dari pemakai sistem, dilakukan tes sistem secara menyeluruh.

4. Implementasi Sistem

Tahap ini adalah prosedur yang dilakukan untuk menyelesaikan desain sistem yang ada dalam dokumen desain sistem yang disetujui, dan menguji, menginstal dan memulai penggunaan sistem baru atau sistem yang diperbaiki. Tujuan dari tahap implementasi ini adalah untuk menyelesaikan

desain sistem yang sudah disetujui, menguji serta mendokumentasikan program-program dan prosedur sistem yang diperlukan.

5. Pemeliharaan Sistem

Tujuan dari pemeliharaan sistem adalah untuk meyakinkan apakah sistem tersebut berjalan sesuai dengan tujuan semula dan apakah masih ada perbaikan atau penyempurnaan yang harus dilakukan. Selain itu tahap ini juga merupakan bentuk evaluasi untuk memantau supaya sistem informasi yang dioperasikan dapat berjalan secara optimal dan sesuai dengan harapan pemakai maupun organisasi yang menggunakan sistem tersebut.

2.2.4 Alat Dalam Pengembangan Sistem

2.2.4.1 Diagram Konteks

Diagram konteks adalah diagram arus data untuk menggambarkan sistem dan lingkungan luar sistem yang saling berhubungan. Diagram Konteks ini adalah bagian dari level tertinggi dari DFD (*Data Flow Diagram*) yang menggambarkan seluruh *input* ke suatu sistem atau *output* dari sistem.

2.2.4.2 *Data Flow Diagram (DFD)*

Data Flow Diagram (DFD) merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur (*structured Analysis and design*). DFD merupakan alat yang cukup populer sekarang ini, karena dapat menggambarkan arus data di dalam sistem dengan terstruktur dan jelas (Jogiyanto, 2005).

Menurut Jogiyanto (2005) elemen dasar dari diagram aliran data adalah sebagai berikut:

1. Kesatuan Luar (*External Entity*)

Sesuatu yang berada di luar sistem, tetapi ia tidak memberikan data ke dalam sistem atau memberikan data dari sistem, disimbolkan dengan suatu

kotak notasi. *External Entity* tidak termasuk bagian (departemen) maka bagian lain yang masih terkait menjadi *external entity*.

Pedoman pemberian nama kesatuan luar (*External Entity*) yaitu:

- Nama terminal berupa kata benda.
- Terminal tidak boleh memiliki nama yang sama kecuali memang objeknya sama (digambarkan dua kali, dimaksudkan untuk membuat diagram lebih jelas). Bila demikian, maka terminal ini perlu di beri garis miring pada pojok kiri atas.

Simbol: notasi kotak



Gambar 2. 2 Simbol Kesatuan Luar

(Sumber: Jogiyanto, 2005)

2. Arus Data (*Data Flow*)

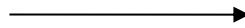
Arus data merupakan tempat mengalirnya informasi dan digambarkan dengan garis yang menghubungkan komponen dari sistem. Arus data ditunjukkan dengan arah panah dan garis diberi nama atas arus data yang mengalir. Arus data ini mengalir di antara proses, data store dan menunjukkan arus data dari data yang berupa masukkan untuk sistem atau hasil proses sistem.

Pedoman pemberian nama aliran data yaitu:

- Nama aliran data yang terdiri dari beberapa aliran kata dihubungkan dengan garis sambung.
- Tidak boleh ada aliran data yang namanya sama, dan pemberian nama harus mencerminkan isinya.
- Aliran data yang terdiri dari beberapa elemen dapat dinyatakan dengan grup elemen.

- Hindari penggunaan kata “data” dan “informasi” untuk memberi nama pada aliran data.
- Sedapat mungkin nama aliran data ditulis lengkap.

Simbol: notasi anak panah (arus data sebaiknya diberi nama yang jelas dan mempunyai arti. Nama arus data dituliskan di samping garis panahnya)



Gambar 2. 3 Simbol Arus Data

(Sumber: Jogiyanto, 2005)

3. Proses (*Process*)

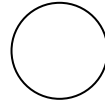
Proses merupakan apa yang dikerjakan oleh sistem. Proses dapat mengolah data atau aliran data masuk menjadi aliran data keluar. Proses berfungsi mentransformasikan satu atau beberapa data masukan menjadi satu beberapa data keluaran sesuai dengan spesifikasi yang diinginkan. Setiap proses memiliki satu atau beberapa masukan serta menghasilkan satu atau beberapa data keluaran. Proses sering pula disebut *bubble*.

Pedoman pemberian nama proses, yaitu:

- Nama proses terdiri dari kata kerja dan kata benda yang mencerminkan fungsi proses tersebut, misalnya Hitung Bonus, Pendataan Karyawan, Cetak Faktur, dan lain-lain.
- Jangan Menggunakan kata proses sebagai bagian dari nama suatu proses (*bubble*).
- Tidak boleh ada beberapa proses yang memiliki nama yang sama.
- Proses harus diberi nomor. Urutan nomor sedapat mungkin mengikuti aliran atau urutan proses, namun demikian urutan proses secara kronologis.

- Penomoran proses pada tingkat pertama (Diagram Nol) adalah 1.0, 2.0, 3.0 dan seterusnya.
- *Context Diagram* tidak perlu diberi nomor.

Simbol: notasi lingkaran



Gambar 2. 4 Simbol Proses

(Sumber: Jogiyanto, 2005)

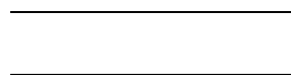
4. Simpanan Data (*Data Store*)

Simpanan data merupakan tempat penyimpanan data pengikat yang ada dalam sistem. Data store dapat disimbolkan dengan sepasang dua garis sejajar atau dua garis dengan salah satu sisi samping terbuka. Proses dapat mengambil data dari atau memberikan data ke *database*.

Pedoman pemberian nama data store, yaitu:

- Nama harus mencerminkan *data store* tersebut.
- Bila namanya lebih dari satu kata maka harus diberi tanda sambung.

Simbol: simpanan data di DFD



Gambar 2. 5 Simbol Simpanan Data

(Sumber: Jogiyanto, 2005)

Hal-hal yang perlu diperhatikan pada *data flow diagram* (DFD) yang memiliki lebih dari satu level adalah:

- Harus terdapat keseimbangan *input* atau *output* antara satu level dan level berikutnya.
- Keseimbangan antara level 0 dan level 1 dilihat pada *input/output* dari aliran data ke atau dari terminal pada level 0 sedangkan

keseimbangan antara level 1 dan level 2 dilihat pada *input/output* dari aliran data ke atau dari proses yang bersangkutan.

- Nama aliran data, data store dan terminal pada setiap level harus sama, apabila objeknya sama.

Ada sumber buku yang menyatakan terminal tidak perlu digambarkan pada level 1,2 dan seterusnya namun untuk memperjelas diagram, maka sebaiknya terminal tetap digambarkan pada level 1,2 dan seterusnya.

2.2.4.3 Flowchart

Pengertian *flowchart* adalah suatu simbol yang digunakan untuk menggambarkan suatu arus data yang berhubungan dengan suatu sistem transaksi akuntansi.

Menurut Jogiyanto (2005), terdapat lima macam bagan alir, yaitu sebagai berikut:

1. Bagan Alir Sistem (*Systems Flowchart*)

Merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan. Bagan ini menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem.

2. Bagan Alir Dokumen (*Document Flowchart*)

Bagan alir dokumen atau disebut juga bagan alir formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya. Bagan alir dokumen ini menggunakan simbol-simbol yang sama dengan yang digunakan di dalam bagan alir sistem.

3. Bagan Alir Skematik (*Schematic Flowchart*)

Merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu untuk menggambarkan prosedur di dalam sistem. Perbedaannya adalah, bagan alir skematis selain menggunakan simbol-simbol bagan alir sistem juga menggunakan gambar-gambar komputer dan peralatan lainnya yang digunakan. Maksud penggunaan gambar-gambar ini adalah untuk

memudahkan komunikasi kepada orang yang kurang paham dengan simbol-simbol bagan alir. Penggunaan gambar-gambar ini memudahkan untuk dipahami, tetapi sulit dan lama menggambarinya.

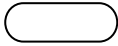

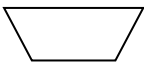
4. Bagan Alir Program (*Program Flowchart*)

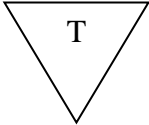

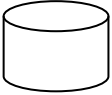




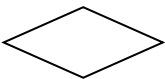

Merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program dapat terdiri dari dua macam, yaitu Bagan Alir Logika Program (*Program Logic Flowchart*) dan Bagan Alir Program Komputer terinci (*Detailed Computer Program Flowchart*).

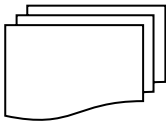
5. Bagan Alir Proses (*Process Flowchart*)

Merupakan bagan alir yang banyak digunakan di teknik industri. Bagan alir ini juga berguna bagi analisis sistem untuk menggambarkan proses dalam suatu prosedur. Bagan alir proses selain dapat menunjukkan kegiatan dan simpanan yang digunakan dalam suatu prosedur, dapat juga menunjukkan jarak kegiatan yang satu dengan yang lainnya serta waktu yang diperlukan oleh suatu kegiatan.

Simbol dari bagan alir (*flowchart*) adalah sebagai berikut ini :

No	Simbol	Pengertian	Keterangan
1.		Mulai / berakhir (<i>Terminal</i>)	Digunakan untuk memulai, mengakhiri, atau titik henti dalam sebuah proses atau program; juga digunakan untuk menunjukkan pihak eksternal.
2.		Dokumen	Sebuah dokumen atau laporan; dokumen dapat dibuat dengan tangan atau dicetak oleh komputer.
3.		Kegiatan Manual	Sebuah kegiatan pemrosesan yang dilaksanakan secara manual.

4.		Arsip	Arsip dokumen disimpan dan diambil secara manual. Huruf di dalamnya menunjukkan cara pengurutan arsip: N = Urut Nomor; A = Urut Abjad = Urut Tanggal.
5.		Input / Output; Jurnal / Buku Besar	Digunakan untuk menggambarkan berbagai media input dan output dalam sebuah bagan alir program.
6.		Penyimpanan	Data disimpan secara permanen pada disk bermagnet.
7.		Penghubung Pada Halaman Berbeda	Menghubungkan bagan alir yang berada di halaman yang berbeda.
8.		Pemasukan Data On Line	Entri data alat oleh on line seperti terminal CRT dan komputer pribadi.
9.		Pemrosesan Komputer	Sebuah fungsi pemrosesan yang dilaksanakan oleh komputer biasanya menghasilkan perubahan terhadap data atau informasi
10.		Arus Dokumen atau Pemrosesan	Arus dokumen atau pemrosesan; arus normal adalah ke kanan atau ke bawah.
11.		Keputusan	Sebuah tahap pembuatan keputusan
12.		Penghubung Dalam Sebuah Halaman	Menghubungkan bagan alir yang berada pada halaman yang sama.

13.		Dokumen Rangkap	Digambarkan dengan menumpuk simbol dokumen dan pencetakan nomor dokumen di bagian depan dokumen pada bagian kiri atas.
-----	---	-----------------	--

Tabel 2. 2 Simbol Flowchart

(Sumber: Jogiyanto, 2005)

2.2.5 Pengolahan Data

Sistem pengolahan data merupakan suatu sistem di dalam perusahaan yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasional, bersifat manajerial dan kegiatan strategi dari suatu organisasi.

2.2.5.1 Pengertian Basis Data

Basis Data (*Database*) adalah kumpulan dari data yang saling berhubungan satu dengan lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Database merupakan salah satu komponen yang penting dalam sistem informasi, karena merupakan basis dalam menyediakan informasi bagi para pemakai (Jogiyanto, 2005).

Basis Data adalah suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi. Basis Data dimaksudkan untuk mengatasi problem pada sistem yang memakai pendataan berbasis berkas.

Basis Data mempunyai prinsip utama yaitu pengaturan data arsip. Dan tujuan utamanya adalah memberi kemudahan dan kecepatan dalam pengambilan kembali data atau arsip. Secara lebih lengkap, pemanfaatan basis data dilakukan untuk memenuhi sejumlah tujuan (objektif) seperti berikut ini :

1. Kecepatan dan kemudahan (*Speed*)

Pemanfaatan basis data memungkinkan kita untuk dapat menyimpan data atau melakukan perubahan atau manipulasi terhadap data atau menampilkan kembali data tersebut dengan lebih cepat dan mudah.

2. Efisiensi ruang penyimpanan (*Space*)

Karena keterkaitannya yang erat antar kelompok data dalam sebuah basis data, maka redundansi (perulangan) pasti selalu ada. Banyaknya redundansi ini pasti akan memperbesar ruang penyimpanan yang harus disediakan. Tetapi dalam basis data hal ini dapat diatasi, yaitu dengan memberikan sejumlah pengkodean atau dengan membuat relasi-relasi.

3. Keakuratan (*Accuracy*)

Pemanfaatan pengkodean atau pembentukan relasi antar data bersama dengan penerapan aturan atau batasan (*constraint*) tipe data, keunikan data, dan sebagainya, yang secara ketat dapat diterapkan yang secara ketat dapat diterapkan dalam sebuah basis data, sangat berguna untuk menekan ke tidak akuratan pemasukan atau penyimpanan.

4. Ketersediaan (*Availability*)

Pertumbuhan data (baik dari jumlah atau jenisnya) sejalan dengan waktu akan semakin membutuhkan ruang penyimpanan yang besar. Padahal tidak semua data itu selalu kita butuh kan atau gunakan. Karena itu kita dapat memilah adanya data utama/master/referensi, data transaksi, data histori hingga data kadaluwarsa.

5. Kelengkapan (*Completeness*)

Lengkap atau tidaknya data yang kita kelola dalam sebuah basis data bersifat relatif (baik terhadap kebutuhan pemakai maupun terhadap waktu). Bila seseorang pemakai sudah menganggap bahwa data yang dipelihara sudah lengkap, maka pemakai lain belum tentu sependapat. Atau, yang sekarang dianggap lengkap, belum tentu dimasa akan datang akan demikian. Oleh karena itu untuk mengakomodasi kebutuhan kelengkapan data yang

semakin berkembang, maka kita tidak hanya menambah record-record data, tetapi juga dapat melakukan perubahan struktur dalam basis data, baik dalam melakukan penambahan objek baru (tabel) atau dengan penambahan field-field baru pada suatu tabel baru.

6. Keamanan (*Security*)

Dalam sistem yang besar maka diperlukan keamanan yang ekstra ketat, hal ini ditujukan untuk melindungi data-data yang dianggap penting, Dalam hal ini basis data dapat memberikan fasilitas keamanan, di antaranya dalam suatu sistem dapat ditetapkan siapa-siapa pemakai yang boleh menggunakan basis data beserta objek-objek di dalamnya dan menentukan jenis-jenis operasi apa saja yang boleh dilakukan.

7. Kebersamaan Pemakaian (*Sharability*)

Dalam penggunaan datanya basis data dapat digunakan oleh lebih dari satu orang, oleh karena itu fungsi dari basis data adalah menjaga atau menghindari terhadap munculnya suatu persoalan baru seperti inkonsistensi data (karena data yang sama diubah oleh banyak orang atau pemakai pada saat yang sama).

2.2.5.2 Pengertian Sistem Basis Data

Menurut Jogiyanto (2005), sistem basis data adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang berhubungan satu dengan yang lainnya dan membuatnya tersedia untuk beberapa aplikasi yang bermacam-macam di dalam suatu organisasi.

Komponen-komponen utama sebuah sistem basis data :

1. Perangkat Keras (*Hardware*)
2. Sistem Operasi (*Operating System*)
3. Basis Data (*Data Base*)
4. Sistem Pengolahan Basis Data

5. Pemakai (*User*)
6. Aplikasi (perangkat lunak) lain (bersifat opsional)

Database dapat dinyatakan sebagai suatu sistem yang memiliki karakteristik sebagai berikut:

1. Merupakan suatu kumpulan "*Interlated Data*" yang disimpan bersama-sama tanpa mengganggu satu sama lain atau membentuk kerangka data.
2. Kumpulan data dalam database yang dapat digunakan oleh sebuah program aplikasi.
3. Data disimpan sedemikian rupa, sehingga bebas dari program aplikasi.
4. Penambahan data baru, modifikasi, dan pengambilan kembali dari data dapat dilakukan dengan mudah dan terkontrol.

2.2.5.3 Database Management System (DBMS)

Database Management System (DBMS) adalah paket perangkat lunak yang kompleks yang digunakan untuk memanipulasi database. Semua operasi *input* dan *output* yang berhubungan dengan database harus menggunakan DBMS. Bila pemakai akan mengakses database, DBMS menyediakan penghubung (*Interface*) antara pemakai dengan database. Hubungan pemakai dengan database dapat dilakukan dengan dua cara:

1. Secara interaktif menggunakan bahasa pertanyaan (*Query language*).
2. Dengan menggunakan program aplikasi.

Dengan menggunakan *query language*, maka pemakai dapat menggunakan database tanpa harus membuat suatu program terlebih dahulu. Hanya dengan perintah-perintah pendek saja, maka dapat berhubungan dengan database. Dengan demikian mereka yang tidak menguasai pemrograman masih memungkinkan untuk memanipulasi database secara langsung dengan mudah. *Query language* biasanya digunakan untuk permintaan informasi yang mendadak. Bila pemrosesan data sifatnya rutin, maka dapat dibuatkan suatu program aplikasi untuk mengakses

database. DBMS mempunyai paling sedikit dua buah komponen, yaitu DDL (*Data Defenition Language*) dan DML (*Data Manipulation Language*).

2.2.5.4 Relational Database Management System (RDBMS)

Relational Database Management System (RDBMS) adalah database di mana tabel-tabel yang ada di dalamnya berelasi satu dengan yang lainnya. Contoh RDBMS adalah interbase, Microsoft Access, Sybase, SQL Server, Oracle Database, Dll.


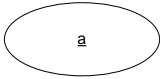
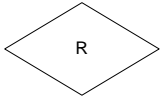

2.2.5.5 Entity Relationship Diagram (ERD)

Entity Relationship Diagram merupakan suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak (Ladjamudin, 2005).

Dari definisi di atas maka dapat disimpulkan bahwa pengertian ERD adalah suatu model jaringan yang menggunakan suatu susunan data untuk menggambarkan hubungan antar penyimpanan atau data.

Notasi-notasi simbolik di dalam diagram E-R yang dapat kita gunakan adalah sebagai berikut:

1. Persegi panjang. Menyatakan himpunan entitas/entitas.
2. Lingkaran/elips. Menyatakan atribut (atribut yang berfungsi sebagai *key* digaris bawah).
3. Belah ketupat. Menyatakan himpunan relasi/relasi.
4. Garis. Sebagai penghubung antara himpunan relasi dengan himpunan entitas dan himpunan entitas dengan atributnya.
5. Kardinalitas relasi dapat dinyatakan dengan banyaknya garis cabang atau dengan pemakaian angka (1 dan 1 untuk relasi satu ke satu, 1 dan N untuk relasi satu ke banyak, atau N dan N untuk relasi banyak ke banyak).

No	Simbol	Keterangan
1		Himpunan entitas E
2		Atribut a sebagai <i>key</i>
3		Himpunan relasi R
4		Link

Tabel 2. 3 Notasi ERD

(Sumber: Ladjamudin, 2005)

2.2.5.6 Kunci Elemen Data (*Key*)

Key adalah elemen *record* yang dipakai untuk menemukan *record* tersebut pada waktu akses, atau bisa juga digunakan untuk mengidentifikasi setiap *entity/record*/baris.

Ada 6 macam *Key* yang dapat diterapkan pada suatu tabel, yaitu :

1. *Superkey*

Superkey merupakan satu atau lebih atribut (kumpulan atribut) dari suatu tabel yang dapat digunakan untuk mengidentifikasi *entity/record* dari tabel tersebut secara unik (tidak semua atribut dapat menjadi *superkey*).

2. *Candidate key*

Superkey dengan jumlah atribut minimal disebut dengan *candidate key*. *Candidate key* tidak boleh berisi atribut dari tabel lain sehingga *candidate key* sudah pasti *superkey* namun belum tentu sebaliknya.

3. *Primary Key*

Salah satu atribut dari *candidate key* dapat dipilih/ditentukan menjadi *primary key* dengan tiga kriteria, yaitu: *key* tersebut lebih natural untuk digunakan sebagai acuan, *Key* tersebut lebih sederhana dan unik.

4. *Alternate key*

Setiap atribut dari *candidate key* yang tidak terpilih menjadi *primary key*, maka atribut-atribut tersebut dinamakan *alternate key*.

5. *Foreign key*

Foreign key merupakan sembarang atribut yang menunjuk kepada *primary key* pada tabel lain. *Foreign key* akan terjadi pada suatu relasi yang memiliki kardinalitas *one to many* (satu ke banyak) atau *many to many* (banyak ke banyak). *Foreign key* biasanya selalu diletakkan pada tabel/relasi yang mengarah ke banyak.

6. *External key*

External key merupakan suatu *lexical attribute* (atau himpunan *lexical attribute*) yang nilai-nilainya selalu mengidentifikasi satu *object instance*.

2.2.5.7 **Kardinalitas**

Relationship degree atau Derajat Relasi adalah jumlah entitas yang berpartisipasi dalam satu relationship. Derajat Relationship yang sering dipakai di dalam ERD :

1. *Unary Relationship*

Unary Relationship adalah model relationship yang terjadi diantara entity yang berasal dari *entity set* yang sama. Sering juga disebut sebagai *Recursive Relationship* atau *Reflective Relationship*.

2. *Binary Relationship*

Binary Relationship adalah model Relationship antara *instance-instance* dari suatu tipe entitas (dua *entity* yang berasal dari *entity* yang sama). Relationship ini paling umum digunakan dalam pembuatan model data.

3. *Ternary Relationship*

Ternary Relationship merupakan relationship antara *instance-instance* dari tiga tipe entitas secara sepihak. Pada gambar di bawah ini relationship mengirimkan mencatat jumlah suatu gudang yang telah ditentukan. Masing-masing entitas mungkin berpartisipasi satu atau banyak dalam suatu relationship *ternary*. Perlu dicatat bahwa relationship ternary tidak sama dengan tiga relationship binary.

2.2.5.8 Derajat Relasi (*Relationship Degree*)

Relationship degree atau Derajat Relasi adalah jumlah entitas yang berpartisipasi dalam satu relationship. Derajat Relationship yang sering dipakai di dalam ERD :

1. *Unary Relationship*

Unary Relationship adalah model relationship yang terjadi diantara entity yang berasal dari *entity* set yang sama. Sering juga disebut sebagai *Recursive Relationship* atau *Reflective Relationship*.

2. *Binary Relationship*

Binary Relationship adalah model Relationship antara *instance-instance* dari suatu tipe entitas (dua *entity* yang berasal dari *entity* yang sama). Relationship ini paling umum digunakan dalam pembuatan model data.

3. *Ternary Relationship*

Ternary Relationship merupakan relationship antara *instance-instance* dari tiga tipe entitas secara sepihak. Pada gambar di bawah ini relationship mengirimkan mencatat jumlah suatu gudang yang telah ditentukan. Masing-masing entitas mungkin berpartisipasi satu atau banyak dalam suatu

relationship *ternary*. Perlu dicatat bahwa relationship ternary tidak sama dengan tiga relationship binary.

2.2.5.9 Kamus Data

Menurut Jogiyanto (2005), kamus data (KD) atau data dictionary (DD) atau disebut juga dengan istilah sistem data dictionary adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi.

Dari definisi di atas maka dapat disimpulkan bahwa pengertian Kamus Data adalah suatu aplikasi yang digunakan sebelum data diaplikasikan atau diakses. Isi dari kamus data adalah sebagai berikut:

1. Nama Arus Data

Karena KD dibuat berdasarkan arus data yang mengalir di DAD, maka nama dari arus juga harus dicatat di KD, sehingga mereka yang membaca DAD dan memerlukan penjelasan lebih lanjut tentang suatu arus data tertentu di DAD dapat langsung mencarinya dengan mudah di KD.

2. Alias

Alias perlu ditulis karena data yang sama mempunyai nama yang berbeda untuk orang atau departemen satu dengan lainnya.

3. Bentuk Data

Digunakan untuk mengelompokkan KD ke dalam kegunaannya sewaktu perancangan sistem.

4. Arus Data

Arus data menunjukkan dari mana data mengalir dan ke mana data akan menuju.

5. Penjelasan

Untuk lebih memperjelas lagi tentang makna dari arus data yang dicatat di KD, maka bagian penjelasan dapat diisi dengan keterangan-keterangan tentang arus data tersebut.

6. Periode

Periode ini menunjukkan kapan terjadinya arus data ini.

7. Volume

Volume yang perlu dicatat di KD adalah tentang volume rata-rata dan volume puncak dari arus data.

8. Struktur Data

Struktur data menunjukkan arus data yang dicatat di KD terdiri dari item-item data apa saja.

2.3 Website

Situs web (bahasa Inggris: *website*) atau yang sering kita kenal dengan istilah situs adalah sejumlah halaman web yang memiliki topik saling terkait, terkadang disertai pula dengan berkas-berkas gambar, video, atau jenis-jenis berkas lainnya. Sebuah situs web biasanya ditempatkan setidaknya pada sebuah server web yang dapat diakses melalui jaringan seperti internet, ataupun jaringan wilayah lokal (LAN) melalui alamat internet yang dikenali sebagai URL. Gabungan atas semua situs yang dapat diakses publik di internet disebut pula sebagai World Wide Web atau lebih dikenal dengan singkatan WWW. Sebuah halaman web merupakan berkas yang ditulis sebagai berkas teks biasa (*plain text*) yang diatur dan dikombinasikan sedemikian rupa dengan instruksi-instruksi berbasis HTML, atau XHTML, kadang-kadang pula disisipi dengan sekelumit bahasa skrip. Berkas tersebut kemudian diterjemahkan oleh peramban web dan ditampilkan seperti layaknya sebuah halaman pada monitor komputer. Halaman-halaman web tersebut diakses oleh pengguna melalui protokol komunikasi jaringan yang disebut sebagai HTTP, sebagai tambahan untuk meningkatkan aspek keamanan dan aspek privasi yang lebih baik, situs

web dapat pula mengimplementasikan mekanisme pengaksesan melalui protokol HTTPS.

Sebuah situs web biasanya ditempatkan setidaknya pada sebuah server web yang dapat diakses melalui jaringan seperti internet, ataupun jaringan wilayah lokal (LAN) melalui alamat internet yang dikenali sebagai URL. Gabungan atas semua situs yang dapat diakses publik di internet disebut pula sebagai World Wide Web atau lebih dikenal dengan singkatan WWW. Sebuah halaman web merupakan berkas yang ditulis sebagai berkas teks biasa (*plain text*) yang diatur dan dikombinasikan sedemikian rupa dengan instruksi-instruksi berbasis HTML, atau XHTML, kadang-kadang pula disisipi dengan sekelumit bahasa skrip. Berkas tersebut kemudian diterjemahkan oleh peramban web dan ditampilkan seperti layaknya sebuah halaman pada monitor komputer. Halaman-halaman web tersebut diakses oleh pengguna melalui protokol komunikasi jaringan yang disebut sebagai HTTP, sebagai tambahan untuk meningkatkan aspek keamanan dan aspek privasi yang lebih baik, situs web dapat pula mengimplementasikan mekanisme pengaksesan melalui protokol HTTPS (Ferdinandus, 2012).

2.4 Pemrograman Website

Penelitian ini membangun laboratorium virtual untuk mata kuliah pemrograman website yang ada di IIB Darmajaya, yaitu :

1. PHP

PHP merupakan bahasa yang hanya dapat berjalan pada server dan hasilnya dapat ditampilkan pada *client*. PHP adalah produk *open source* yang dapat digunakan secara gratis tanpa harus membayar untuk menggunakannya. Interpreter PHP dalam mengeksekusi kode PHP pada sisi server (disebut *server-side*), sedangkan tanpa adanya interpreter PHP, maka semua skrip dan aplikasi PHP yang dibuat tidak dapat dijalankan (Nugroho, 2008).

2. HTML

HyperText Markup Language (HTML) adalah bahasa yang digunakan untuk menulis halaman web. HTML merupakan pengembangan dari standar pemformatan dokumen teks, yaitu *Standard Generalized Markup Language* (SGML). HTML Pada dasarnya merupakan dokumen ASCII atau teks biasa yang dirancang untuk tidak tergantung pada suatu sistem operasi tertentu (Suryana, 2014).

HTML dokumen tersebut mirip dengan dokumen tulisan biasa, hanya dalam dokumen ini sebuah tulisan bisa memuat instruksi yang ditandai dengan kode atau lebih dikenal dengan TAG tertentu. Sebagai contoh jika ingin membuat tulisan ditampilkan menjadi tebal seperti: TAMPIL TEBAL, maka penulisannya dilakukan dengan cara: `< b> TAMPIL TEBAL`. Tanda `< b>` digunakan untuk mengaktifkan instruksi cetak tebal, diikuti oleh tulisan yang ingin ditebalkan, dan diakhiri dengan tanda `` untuk menonaktifkan cetak tebal tersebut.

3. CSS

Menurut Suryana (2014), CSS (Cascading Style Sheet) adalah suatu bahasa stylesheet yang digunakan untuk mengatur tampilan suatu website, baik tata letaknya, jenis huruf, warna, dan semua yang berhubungan dengan tampilan. Pada umumnya CSS digunakan untuk menformat halaman web yang ditulis dengan HTML atau XHTML.

4. JavaScript

Java Script adalah bahasa script yang berdasar pada objek yang memperbolehkan pemakai untuk mengendalikan banyak aspek interaksi pemakai pada suatu dokumen HTML. Dimana objek tersebut dapat berupa suatu window, frame, URL, dokumen, form, button atau item yang lain. Semua hal tersebut mempunyai properti yang saling berhubungan dengannya dan masing-masing memiliki nama, lokasi, warna nilai dan atribut lain (Suryana, 2014).

2.5 Perangkat Lunak yang Dipergunakan Untuk Membangun Sistem

Ada tiga perangkat lunak yang digunakan untuk mengembangkan sistem laboratorium virtual, yaitu :

1. Node.js

Menurut McCune (2011), Node.js adalah salah satu *framework* untuk pengembangan web, yang dikembangkan pada tahun 2009 oleh Ryan Dahl, Node.js atau (Node) adalah *single-thread server-side JavaScript Environment* yang dibuat dengan bahasa C dan C++. Node.js dirancang agar mudah digunakan yang ekspresif dan fungsionalitas tinggi untuk pemrograman *server-side* yang sangat populer bagi kalangan pengembang *software*. Node.js menggunakan JavaScript *engine* V8 yang dikembangkan oleh Google, cepat dan *powerful* sehingga Node.js merupakan *framework* dengan performa tertinggi. Peneliti menggunakan Node.js sebagai *server-side programming*.

Node.js berisi *framework* yang di dalamnya sudah terdapat web server sehingga kita sudah tidak perlu menggunakan web server lainnya seperti Apache dan IIS. Bahasa pemrograman yang digunakan pada Node.js adalah Javascript. Portal resmi Node.js dapat dikunjungi di alamat website <http://www.nodejs.org>.

2. Apache

Menurut Firdaus (2007), Web server merupakan sebuah bentuk server yang khusus digunakan untuk menyimpan halaman website atau *homepage*. Apache merupakan turunan dari web server yang dikeluarkan oleh NCSA yaitu NCSA HTTPd sekitar tahun 1995-an. Pada dasarnya, Apache adalah “APatCHy” (*patch*) dan pengganti dari NCSA HTTPd. Apache web server merupakan tulang punggung permintaan dari client yang menggunakan browser, seperti Netscape Navigator, Internet Explorer, Mozilla, lynx dan lain-lain. Web Server dalam berkomunikasi dengan kliennya menggunakan protokol HTTP (*Hyper Text Transfer Protocol*). Apache berada di bawah GNU, *General Public License* yang bersifat *free* sehingga Apache dapat didownload gratis pada alamat <http://www.apache.org>. Adapun pertimbangan dalam memilih Apache adalah:

- a. Apache termasuk dalam kategori *free software* (software gratis).

- b. Instalasi apache sangat mudah.
- c. Mampu beroperasi pada banyak platform sistem operasi, seperti Linux, Windows dan lain-lain.

Apache Web Server merupakan web server yang bersifat *open source* dan mempunyai *Performance* yang sangat bagus, fleksibel dan mendukung berbagai macam platform sistem operasi seperti Windows NT/9x, UNIX, Netware 5x, OS/2 dan berbagai macam sistem operasi lainnya. Apache sangat cepat sekali mengeluarkan versi terbarunya untuk mengurangi munculnya *bugs* dan kelemahan program.

3. MySQL

My Structured Query Language (MySQL) adalah sebuah program pembuat dan pengelola *database* atau yang sering disebut *Database Management System* (DBMS). sifat dari DBMS ini adalah *open source*. MySQL juga merupakan program mengakses database yang bersifat jaringan, sehingga dapat digunakan untuk aplikasi *multi user* (banyak pengguna). Kelebihan lain dari MySQL adalah menggunakan bahasa *query* (permintaan) standar SQL. SQL adalah suatu bahasa permintaan yang terstruktur, SQL telah distandarkan untuk semua program mengakses *database* (Nugroho, 2008).

2.6 Penelitian Terkait

Ciepiela (2010), menerbitkan V-Lab mereka yang disebut GridSpace 2 untuk mendukung pemrograman eksplorasi, di mana masing-masing Eksperimen terdiri dari cuplikan yang diprogram secara interaktif di beberapa bahasa pemrograman menggunakan konsol web.

Li Yuanyuan (2015), menerbitkan karya ilmiah mereka tentang penggunaan V-Lab dalam kurikulum ilmu komputer. Berbagai V-Lab telah dikembangkan di beberapa mata kuliah seperti pengolahan citra digital, digital sirkuit, digital pemrosesan sinyal, dan kriptografi. Mereka menggunakan *applet* java teknologi untuk sisi klien

dan server web Java untuk sisi server. Para siswa diperbolehkan memilih java komponen atau komponen layanan web untuk melakukan percobaan.

V-Lab lainnya digagas pada tahun 2017 oleh Stark (2017). Mereka menggunakan server Node.js yang menjalankan simulasi Pemrograman Matlab melalui web server dan komunikasi disampaikan melalui jaringan socket.io. Pada penelitian ini peneliti mengembangkan laboratorium virtual yang menekankan pada bahasa pemrograman pengembangan web seperti Javascript, HTML, CSS dan PHP. Virtual Lab ini memungkinkan siswa melakukan penulisan kode kapan saja dan di mana saja dan mendapatkan hasil langsung selama mereka terhubung ke internet.